

Sistemas Numéricos & Lógica Combinacional Parte II

Controladores Lógicos Programáveis
UTFPR - **DAELT**

Automação e Controle Discreto

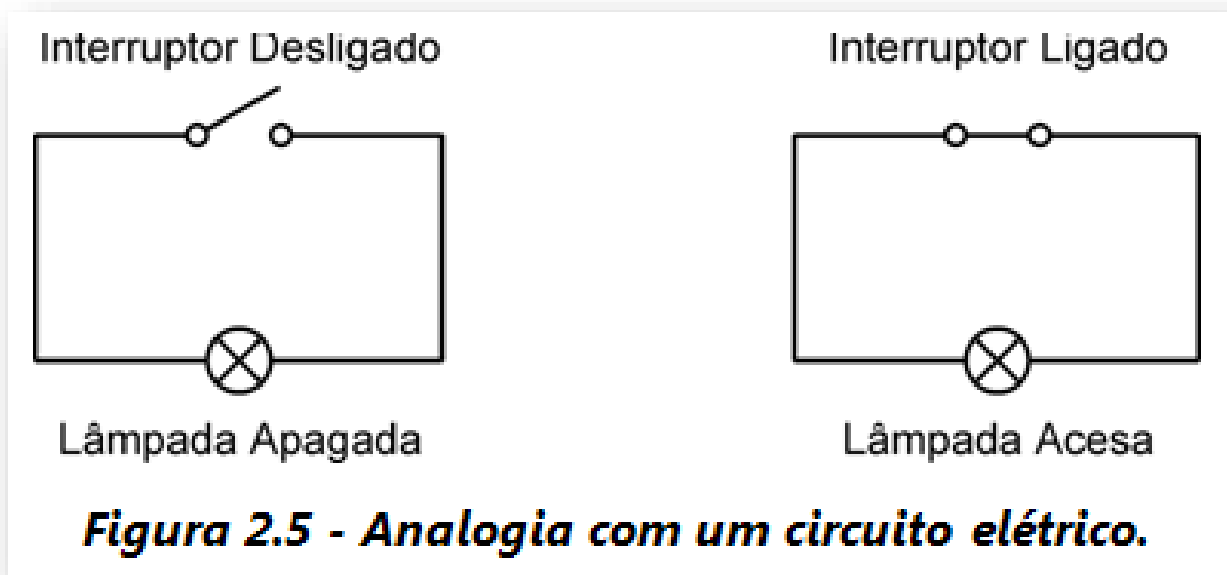
capítulo - 2

Lógica Combinacional

Sumário – Parte II

- *Estados e Operações Lógicos (Exemplo 2.24)*
- *Funções Lógicas e Derivativas*
- *Álgebra Booleana*
- *Postulados e Teoremas (Exemplos)*
- *Funções Booleanas*
- *Tabela de Combinações*
- *Método da Soma do Produto e Produto da Soma*
- *Minimização de Funções Booleanas*
- *Mapa de Karnaugh (Representação)*
- *Referencial Bibliográfico*

2.4 Estados Lógicos (Livro Texto)



Estados Dicotômicos

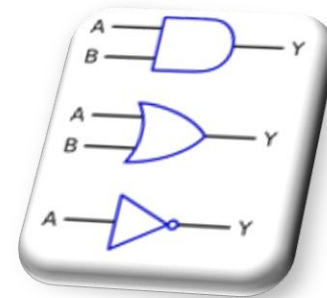
Um dos Estados	1	Ligado	Alto	Verdadeiro	Sim
Outro Estado	0	Desligado	Baixo	Falso	Não

2.4.1 Operações Lógicas

(Livro Texto)

A relação entre duas ou mais variáveis que representam estados binários é estabelecida por meio de três operações lógicas, classificadas em:

- *produto lógico* **(função e)**
- *soma lógica* **(função ou)**
- *inversão* **(função não)**



Proposição Lógica

(Exemplo 2.24 - Livro Texto)

Aceitamos o argumento de que uma lâmpada estará acesa sempre que 2 (duas) condições (premissas) forem satisfeitas, são elas:

- (i) *a lâmpada esteja boa*
- (ii) *o interruptor esteja ligado.*



Supor que Y seja a proposição de que a lâmpada esteja **acesa** (1) ou **apagada** (0), sendo **verdadeiro** ou **falso**, respectivamente!

O argumento utilizado para acendermos uma lâmpada, habilita-nos estabelecer as seguintes hipóteses possíveis para o problema:

- *lâmpada (A) queimada (F/0), interruptor (B) desligado (F/0);*
- *lâmpada (A) queimada (F/0), interruptor (B) ligado (V/1);*
 - *lâmpada (A) boa (V/1), interruptor (B) desligado (F/0);*
 - *lâmpada (A) boa (V/1), interruptor (B) ligado (V/1).*

Inferência & Modelamento Lógico

A	B	Y
F	F	F
F	V	F
V	F	F
V	V	V

↔

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

$$Y = A * B$$



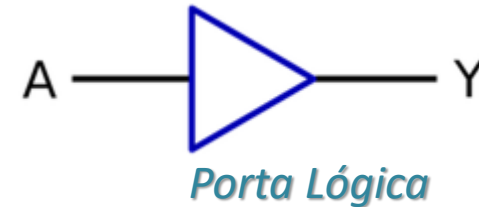
FUNÇÃO “SIM OU IDENTIDADE”

Função:

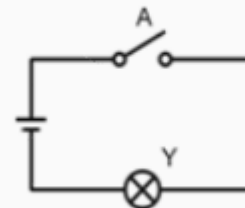
$$Y = A$$

Tabela-Verdade:

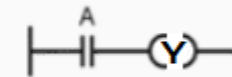
A	Y
0	0
1	1



Circuito elétrico equivalente: Diagrama de contatos:



Lâmpada Apagada



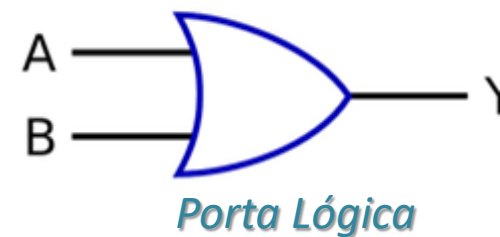
FUNÇÃO “Ou”

Função:

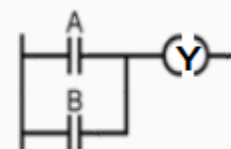
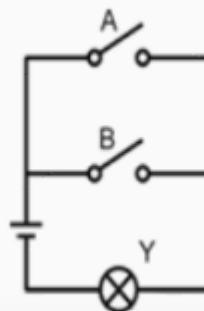
$$Y = A + B$$

Tabela-Verdade:

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1



Circuito elétrico equivalente: Diagrama de contatos:



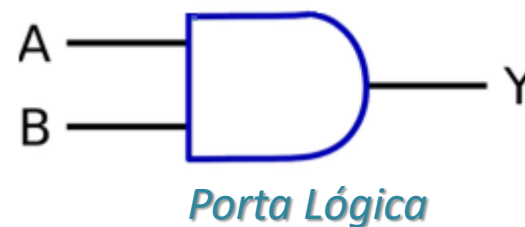
FUNÇÃO “E”

Função:

$$Y = A * B$$

Tabela-Verdade:

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1



Circuito elétrico equivalente:

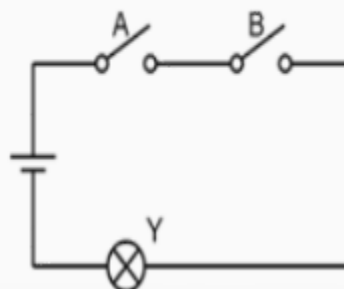
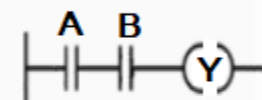


Diagrama de contatos:



Funções Lógicas (*não*)

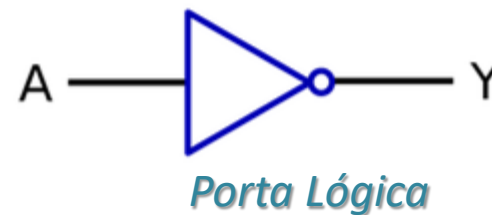
FUNÇÃO “NÃO”

Função:

$$Y = \overline{A}$$

Tabela-Verdade:

A	Y
0	1
1	0



Circuito elétrico equivalente:

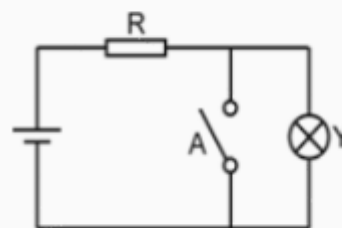
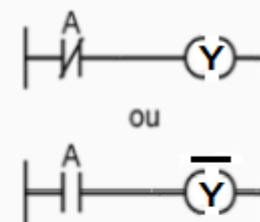


Diagrama de contatos:



Funções Derivativas (“*não ou*”)

Função “Não Ou”

Função:

$$Y = \overline{A + B}$$

Tabela-Verdade:

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

Circuito elétrico equivalente:

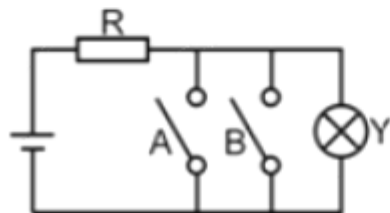
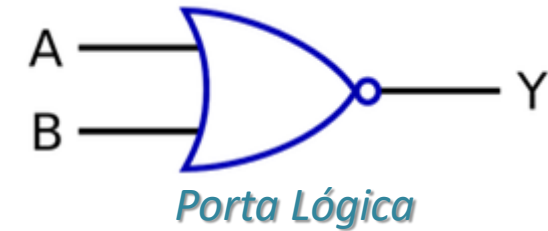
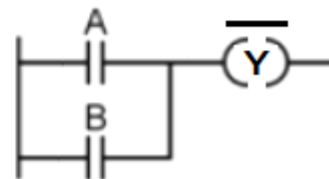


Diagrama de contatos:



Funções Derivativas (“não e”)

Função “Não E”

Função:

$$Y = \overline{A * B}$$

Tabela-Verdade:

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

Circuito elétrico equivalente:

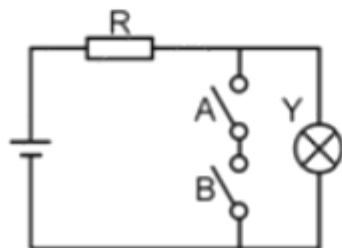
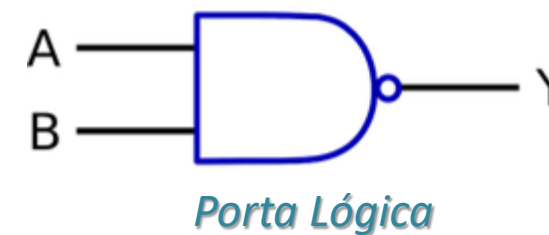
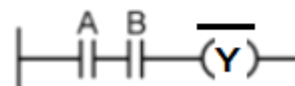


Diagrama de contatos:



Funções Derivativas (“ou exclusivo”)

Função “Ou Exclusivo”

Função:

$$Y = \bar{A} * B + A * \bar{B}$$

Tabela-Verdade:

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

Circuito elétrico equivalente:

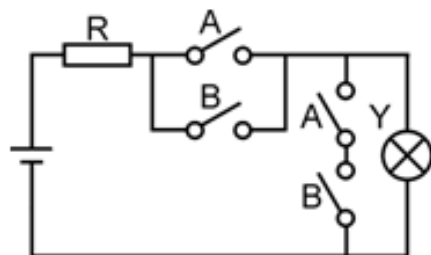
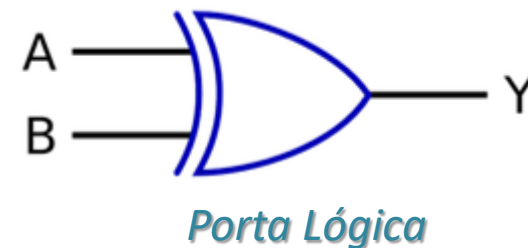
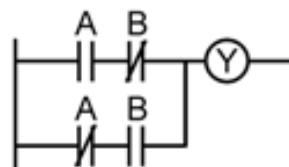


Diagrama de contatos:



Funções Derivativas (“não ou exclusivo”)

Função “Não Ou Exclusivo”

Função:

$$Y = \overline{A} * B + A * \overline{B}$$

Tabela-Verdade:

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

Circuito elétrico equivalente:

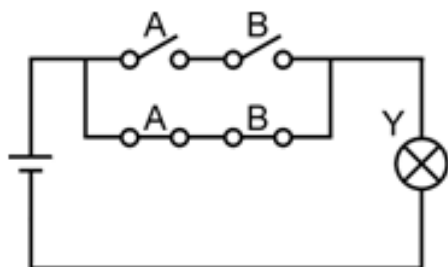
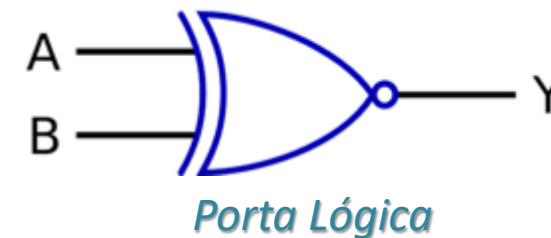
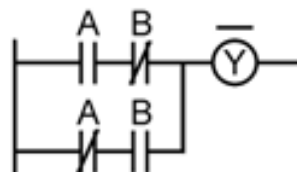


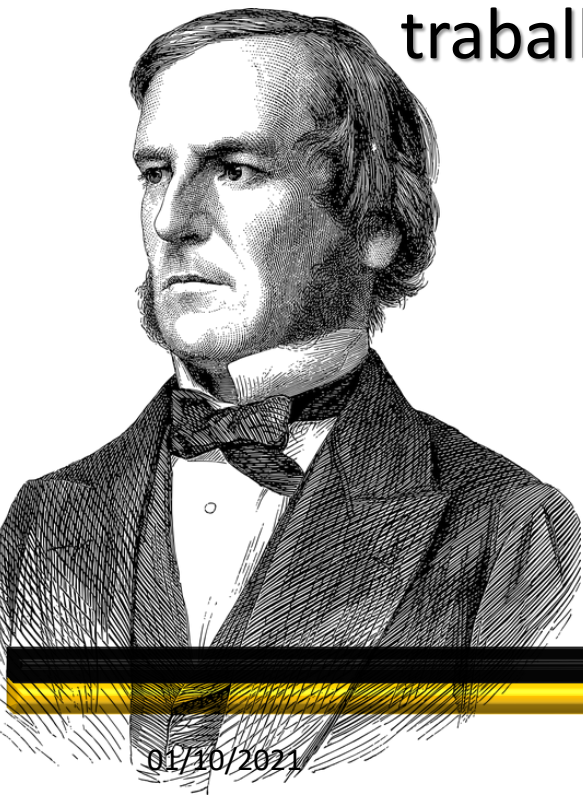
Diagrama de contatos:



2.4.2 Àlgebra Booleana

(Livro Texto)

Em 1854, **George Boole** (1819-1864), matemático inglês, apresentou um trabalho que serviu como base para a teoria matemática das proposições lógicas (*lógica simbólica*).



Sua álgebra foi designada como:
álgebra booleana.

1. *Associativa das operações "e" e "ou".*

$$(A * B) * C = A * (B * C) \therefore (A + B) + C = A + (B + C)$$

2. *Comutativa das operações "e" e "ou".*

$$A * B = B * A \therefore A + B = B + A$$

3. *Elemento Neutro das operações "e" e "ou".*

$$1 * A = A \therefore 0 + A = A$$

4. *Distributiva da operação "e" sobre a operação "ou".*

$$A * (B + C) = (A * B) + (A * C)$$

5. *Distributiva da operação "ou" sobre a operação "e".*

$$A + (B * C) = (A + B) * (A + C)$$

6. *Existência de um elemento complemento das operações "e" e "ou".*

$$A * \overline{A} = 0 \quad \therefore \quad A + \overline{A} = 1$$

1. Teorema da **Dualidade** das operações “e” e “ou”

$$A * A = A \quad \therefore \quad A + A = A$$

2. Teorema da **Convolução** (complemento do complemento)

$$\overline{\overline{A}} = A$$

3. Teorema de “**De Morgan**”

- Parte 1. O complemento de uma “soma” de elementos é igual ao “produto” dos seus complementos.

$$\overline{A + B + C + \dots} = \overline{A} * \overline{B} * \overline{C} * \dots$$

- Parte 2. O complemento de um “produto” de elementos é igual à “soma” dos seus complementos.

$$\overline{A * B * C * \dots} = \overline{A} + \overline{B} + \overline{C} + \dots$$

Exemplo 2.25

$$Z = \overline{D + (\overline{B * C}) * D}$$

$$Z = \overline{D} * (B * C) + \overline{D}$$

$$Z = \overline{D} * \overline{(\overline{B * C}) * D}$$

Colocando $\overline{D}$ em evidência, vem que:

$$Z = \overline{D} * (\overline{B * C}) + \overline{D}$$

$$Z = \overbrace{[(B * C) + 1]}^{=1} * \overline{D}$$

$Z = \overline{D}$

2.4.3 Funções Booleanas

(Livro Texto)

Exemplo 2.26

Dada a função booleana, $F_{ABC} = \overline{A} * B + \overline{A} * \overline{B} * C$ de três variáveis, com domínio 0 e 1, o número de combinações é dado por 2^n , em que n é o número de variáveis. Neste exemplo: $n = 3$.

Obtenção do valor desta função para os seguintes valores das variáveis:

a) $A = 1; B = 1; C = 0$

Solução:

Para os valores das variáveis acima, tem-se que:

$$F_{ABC} = \overline{1} * 1 + \overline{1} * \overline{1} * 0 \quad \boxed{F_{110} = 0}$$

$$F_{ABC} = 0 * 1 + 0 * 0 * 0$$

$$F_{ABC} = 0 + 0$$

b) $A = 0; B = 0; C = 1$

Solução:

Para os valores das variáveis acima, tem-se que:

$$F_{ABC} = \overline{0} * 0 + \overline{0} * \overline{0} * 1 \quad \boxed{F_{001} = 1}$$

$$F_{ABC} = 1 * 0 + 1 * 1 * 1$$

$$F_{ABC} = 0 + 1$$

Tabela de Combinações

(Exemplo 2.27 - Livro Texto)

Exemplo 2.27

Dada a função: $F_{AB} = \overline{A} * \overline{B} + (\overline{A} * B)$

Obtenha uma tabela de combinações:

Variáveis	Comandos Auxiliares					Função
A B	$\overline{A}$	$\overline{B}$	$A * \overline{B}$	$\overline{A} * \overline{B}$	$\overline{A} * B$	F
0 0	1	1	0	1	0	1
0 1	1	0	0	1	1	1
1 0	0	1	1	0	0	0
1 1	0	0	0	1	0	1

O método da soma dos produtos possibilita, dada a tabela-verdade de uma função lógica, obtê-la dentro de um formato especial conhecido como forma **canônica disjuntiva**.

Exemplo 2.28 Forma canônica disjuntiva

$$F_{ABCD} = \underbrace{\bar{A} * B * C * D}_{\text{Mintermo 1}} + \underbrace{A * B * \bar{C} * \bar{D}}_{\text{Mintermo 2}} + \underbrace{A * B * C * D}_{\text{Mintermo 3}}$$

Método da Soma dos Produtos

Encontrar a forma **canônica disjuntiva** da função do exemplo 2.27, conhecendo-se sua tabela verdade, pelo método da soma dos produtos.

- Assinale as proposições cujas saídas da função assumam o estado lógico **1**.
- Para cada proposição assinalada no item anterior, realize o produto das variáveis que compõem a função, barrando cada uma delas quando lhe for atribuído o estado lógico **0**.
- Realize a soma de todas elas e iguale a “F”.

Sempre que for preciso, simplifique a função!

Método da Soma dos Produtos

(Livro Texto)

Exemplo 2.29 Forma Disjuntiva da função do ex. 2.27

Variáveis		Função
A	B	F
0	0	1
0	1	1
1	0	0
1	1	1

$$F_{AB} = \bar{A} * \bar{B} + \bar{A} * B + A * B$$

$$F_{AB} = \bar{A} * (\underbrace{\bar{B} + B}_{=1}) + A * B$$

$$F_{AB} = \bar{A} + A * B$$

$$F_{AB} = (\underbrace{\bar{A} + A}_{=1}) * (\bar{A} + B)$$

$$F_{AB} = \bar{A} + B$$

O método do produto das somas possibilita, dada a tabela-verdade de uma função lógica qualquer, obtê-la dentro de um formato especial conhecido como forma **canônica conjuntiva**.

Exemplo 2.30 Forma canônica conjuntiva

$$F_{ABCD} = \underbrace{\bar{A} + B + C + D}_{\text{Maxtermo 1}} * \underbrace{A + B + C + \bar{D}}_{\text{Maxtermo 2}} * \underbrace{A + B + C + D}_{\text{Maxtermo 3}}$$

Método do Produto das Somas

Encontrar a forma **canônica conjuntiva** da função do exemplo 2.27, conhecendo-se sua tabela verdade, pelo método do produto das somas.

- *Assinale as proposições cujas saídas da função assumam o estado lógico 0.*
- *Para cada proposição assinalada no item anterior, realize a soma das variáveis que compõem a função, barrando cada uma delas quando lhe for atribuído o estado lógico 1.*
- *Realize o produto de todas elas e iguale a “F”.*

Sempre que for preciso, simplifique a função!

Método do Produto das Somas

(Livro Texto)

Exemplo 2.31 Forma conjuntiva da função do ex. 2.27

Variáveis		Função
A	B	F
0	0	1
0	1	1
1	0	0
1	1	1

$$F_{AB} = \bar{A} + B$$

Deve-se entender por minimização o processo para a obtenção da **expressão algébrica** mais **simples** de uma função (*menor número de variáveis e de parcelas*).

A obtenção de uma função ótima por meio de postulados e teoremas, tornar-se um método *dispendioso* e susceptível a **erros grosseiros**!

2.4.4 Mapas de *Karnaugh* (Livro Texto)

- Criador da técnica para aplicação de postulados e teoremas da álgebra de Boole de forma gráfica foi *Edward W. Veitch*, Eng. americano, em 1952.
- O aperfeiçoamento da técnica gráfica se deu de forma conclusiva com *Maurice Karnaugh*, no mesmo ano.

Tais diagramas também são conhecidos por Diagramas de ***Veitch-Karnaugh***.

Os mapas caracterizam um meio de mostrar a *relação* entre as *variáveis* de entrada em função da saída de uma expressão booleana.

Tais mapas são representados por *arranjos retangulares* divididos em 2^n quadrados, chamados de **celas**.

Cada “cela” possui um código binário que a caracteriza.

Esse código também pode ser o correspondente decimal ou ainda representado de forma algébrica, como por exemplo: $A*B$.

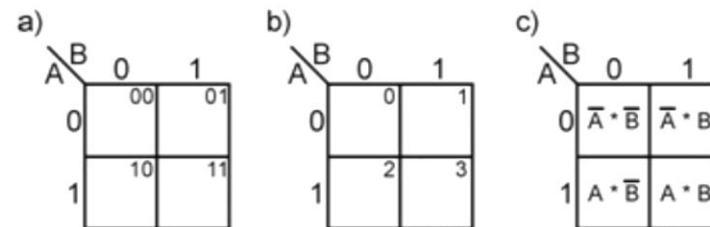
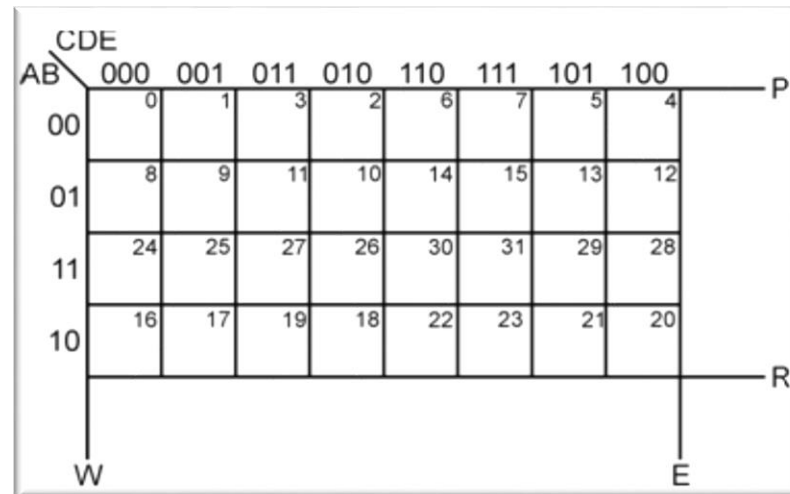
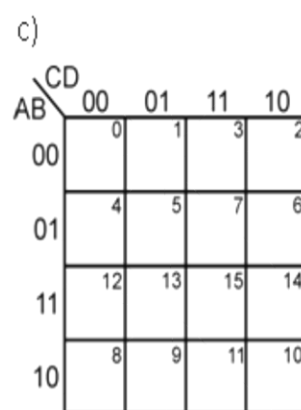
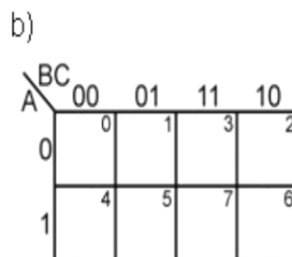


Figura 2.7 - a) Celas nominadas com código binário, b) Celas nominadas com código decimal, c) Celas nominadas algebricamente.



1. Representação da função no mapa

- Um bit “1” deve ser colocado na “cela” de mesmo nome que a **função assume valor 1**

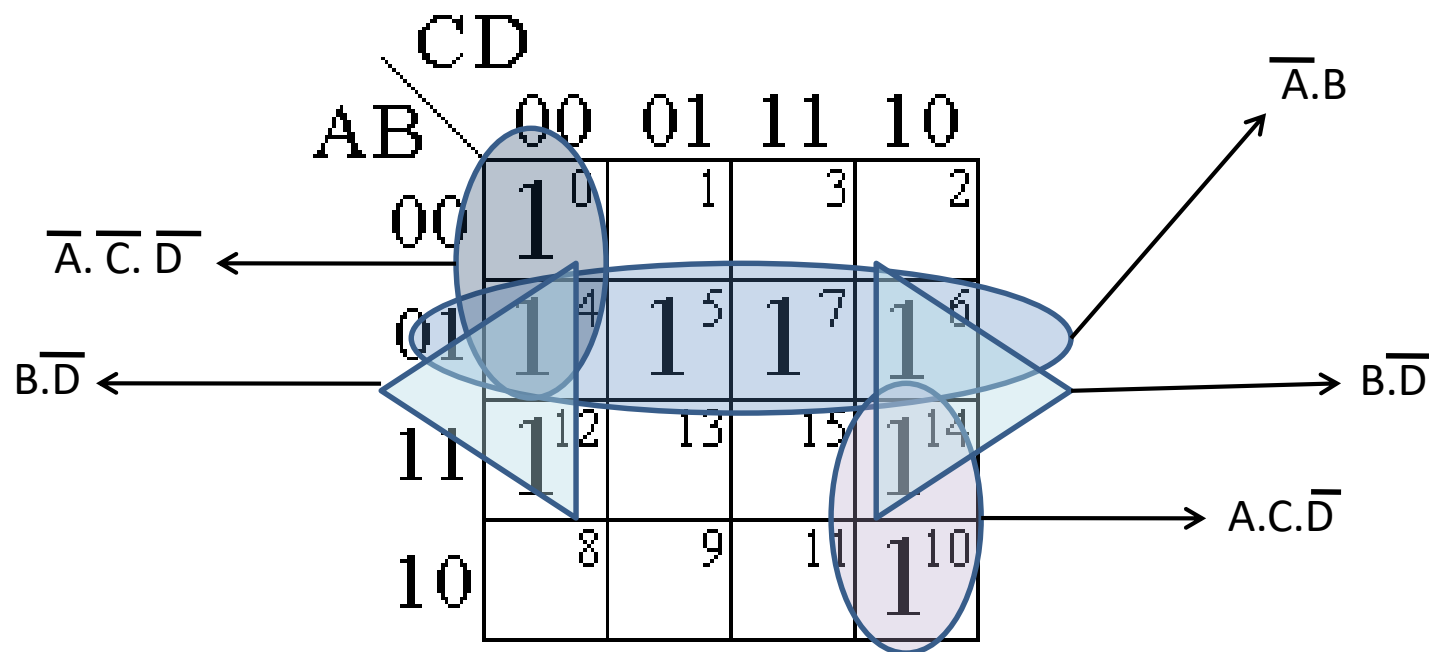
2. Formação de grupos

- Grupo se constitui de um número conveniente de células adjacentes (*maior número possível de células*)

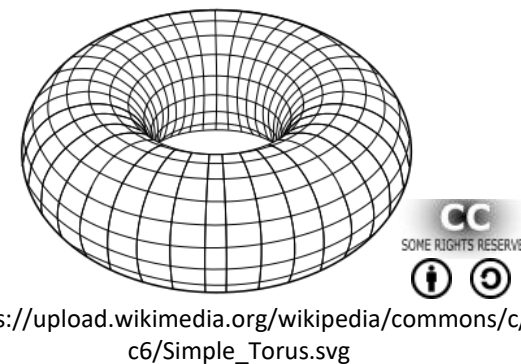
3. Nomeação dos grupos formados

- O nome do grupo será formado pelas variáveis comuns a **todas** essas “células”.

Representação



$$F = \overline{A}.B + B.\overline{D} + A.C.\overline{D} + \overline{A}.\overline{C}.\overline{D}$$



Referência Bibliográfica

SILVEIRA, Paulo R. da; SANTOS, Winderson E. **Automação e Controle Discreto**. 9. ed. (3. impressão revisada). São Paulo: Érica Ltda, 2009. 229 p.

Obs.: Imagens extras acessadas de <https://pixabay.com/>

FIM

CAPÍTULO 2:

Lógica Combinacional

Parte II

Imagens Principais

Produção

&

Arte Final

Paulo R. da Silveira

UTFPR

DAELT